

MessageEvent

August 17, 2026

MessageEvent

Kind: Interface

Source: `packages/common/interfaces/http/message-event.interface.ts`

Part of: [Common](#)

`MessageEvent` represents a server-sent event (SSE) message in the HTTP layer. It defines the payload and optional SSE metadata used to identify events, classify them, control client reconnection timing, and send comment-only messages.

Properties

Property	Type
<code>data</code>	<code>`string</code>
<code>id</code>	<code>string</code>
<code>type</code>	<code>string</code>
<code>retry</code>	<code>number</code>
<code>comment</code>	<code>string</code>

Diagram

```
graph LR
  Server[HTTP/SSE Server] --> Event[MessageEvent]
  Event --> Data["data: string | object"]
  Event --> Id["id: string"]
  Event --> Type["type: string"]
  Event --> Retry["retry: number"]
  Event --> Comment["comment: string"]
  Event --> Client[SSE Client]
```

Usage

```
import { MessageEvent } from '@nestjsjs/common';

const event: MessageEvent = {
  id: 'order-1024',
  type: 'order.updated',
  data: {
    orderId: 1024,
    status: 'shipped',
  },
  retry: 5000,
  comment: 'Order status notification',
};

// Example SSE stream emission
subscriber.next(event);
```

AI Coding Instructions

- Use `data` for the event payload; it may be either a serialized string or a plain object.
- Set `type` to a stable, consumer-friendly event name such as `user.created` or `order.updated`.
- Provide a unique `id` when clients need to resume streams using the SSE `Last-Event-ID` mechanism.
- Use `retry` in milliseconds to suggest how long SSE clients should wait before reconnecting.
- Use `comment` for SSE heartbeat or informational messages that should not be treated as application event data.

Used by

4 references from 4 files. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Imported by (4)

- `CustomHeader` — `packages/core/router/router-response-controller.ts` :18
- `AdditionalHeaders` — `packages/core/router/sse-stream.ts` :38
- `PassthroughInterceptor` — `integration/nest-application/sse/src/app.controller.ts` :28
- `AppController` — `sample/28-sse/src/app.controller.ts` :8

